

بسمه تعالی



آزمایش پنجم آزمایشگاه مدارهای منطقی

استاد حسابی

واحد محاسبات و منطق (ALU)

مریم شیران

دانشگاه صنعتی شریف

بهار ۱۴۰۲

فهرست

۳	مقدمه و هدف
۳	آشنایی با تراشه ۷۴۱۸۱
۳	۱.۱- سیگنال های ورودی و خروجی
۳	۱.۲- تجزیه و تحلیل تیوری آزمایش
۵	۱.۳- نحوه ی ساخت مدار
۶	۱.۴- امتحان اجزای مدار
۸	ساخت مدار داخلی ALU
۸	۲.۱- تجزیه و تحلیل تیوری آزمایش
۸	۲.۲- نحوه ی ساخت مدار
۹	۲.۳- امتحان اجزای مدار
۱۰	منابع و مراجع

مقدمه و هدف

هدف از این آزمایش، آشنایی با واحد محاسبات و منطق (ALU) است. بخش های مختلف این آزمایش در نرم افزار proteus انجام شده است.

آشنایی با تراشه ۷۴۱۸۱

۱.۱- سیگنال های ورودی و خروجی

خطوط داده D3-D0 - خطوط دستور M2-M0 - یک کلید از نوع button-push برای بازگرداندن مدار به حالت اولیه - (Reset) - یک کلید از نوع button-push برای ورودی clock .

این مدار سیگنال خروجی خاصی ندارد. برای بررسی کارکرد درست مدار باید محتویات ثباتهای A و B و خروجی ALU قابل مشاهده باشد.

۱.۲- تجزیه و تحلیل تیوری آزمایش

شکل کلی مدار مانند تصویر در شرح آزمایش میباشد تنها باید قسمت کنترل کننده را به گونه ای بسازیم که مدار حاصل مانند زیر عمل کند :

M2	M1	M0	Operation
0	0	0	$A \leftarrow D_3-D_0$
0	0	1	$B \leftarrow D_3-D_0$
0	1	0	$A \leftarrow A$
0	1	1	$A \leftarrow B$
1	0	0	clear (A)
1	0	1	$A \leftarrow not(A)$
1	1	0	$A \leftarrow and(A,B)$
1	1	1	$A \leftarrow add(A,B)$

برای اینکار به دیتا شیت تراشه ی ۷۴۱۸۱ که به عنوان ALU استفاده میشود نیز نیاز داریم :

SELECTION				M = H LOGIC FUNCTIONS
S3	S2	S1	S0	
L	L	L	L	$F = \overline{A}$
L	L	L	H	$F = \overline{A+B}$
L	L	H	L	$F = \overline{AB}$
L	L	H	H	$F = 0$
L	H	L	L	$F = \overline{AB}$
L	H	L	H	$F = \overline{B}$
L	H	H	L	$F = A \oplus B$
L	H	H	H	$F = A\overline{B}$
H	L	L	L	$F = \overline{A+B}$
H	L	L	H	$F = A \oplus B$
H	L	H	L	$F = B$
H	L	H	H	$F = AB$
H	H	L	L	$F = 1$
H	H	L	H	$F = A + \overline{B}$
H	H	H	L	$F = A + B$
H	H	H	H	$F = A$

لازم به ذکر است که این تنها بخشی از دیتاشیت تراشه به ازای ام برابر ۱ میباشد اما چون همین کافی است از آوردن باقی دیتا شیت خودداری شده است.

حال با استفاده از دو جدول بالا جدول زیر را به دست میآوریم :

M_2	M_1	M_0	S_A	S_B	S_3	S_2	S_1	S_0
0	0	0	1	1	×	×	×	×
0	0	1	0	0	1	1	1	1
0	1	0	0	1	1	1	1	1
0	1	1	0	1	1	0	1	0
1	0	0	0	1	0	0	1	1
1	0	1	0	1	0	0	0	0
1	1	0	0	1	1	0	1	1
1	1	1	0	1	1	1	1	0

اکنون هر یک از سیگنال های کنترلی را برحسب ورودی های M ساده میکنیم.

$$S_A = (m_2 + m_1 + m_0)'$$

$$S_B = m_2 + m_1 + m_0'$$

$$S_0 = (m_1 + m_2)' + m_0'$$

$$S_1 = m_2' + m_1 + m_0'$$

$$S2 = m2 \ m1 \ m0 + m2' \ m1 \ m0' + m2' \ m1' \ m0$$

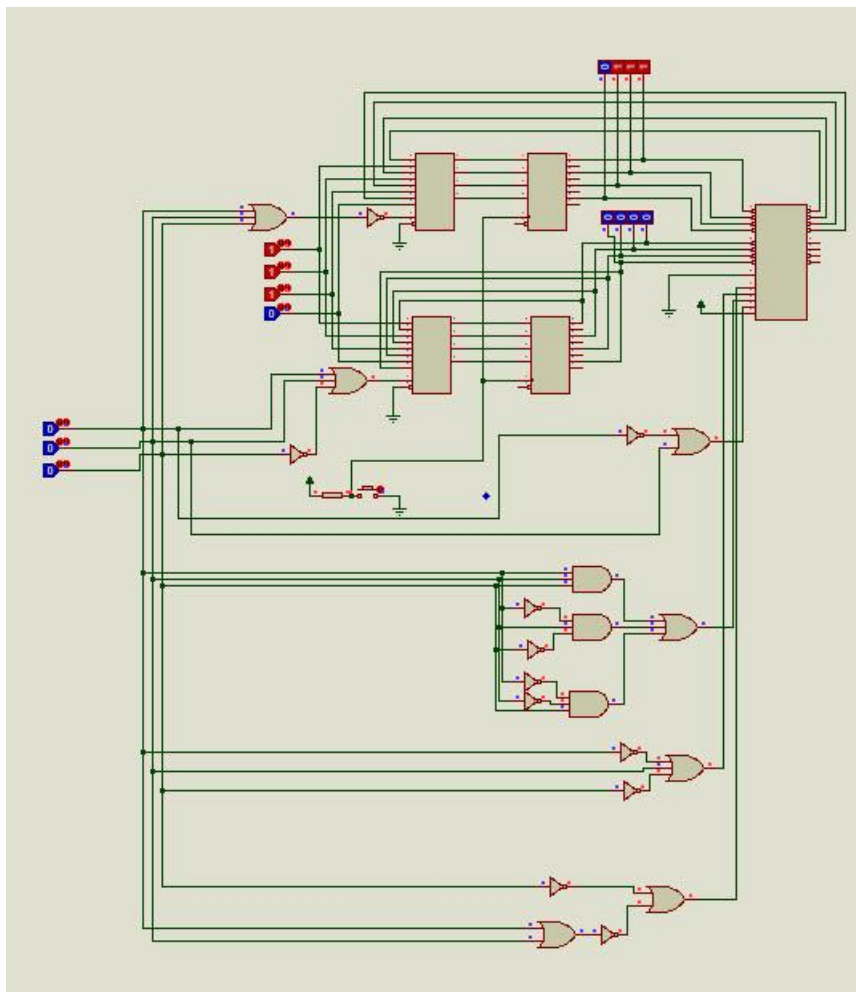
$$S3 = m2' + m1$$

حال مطابق عبارات بالا مدار را میسازیم.

۱.۳ - نحوه ی ساخت مدار

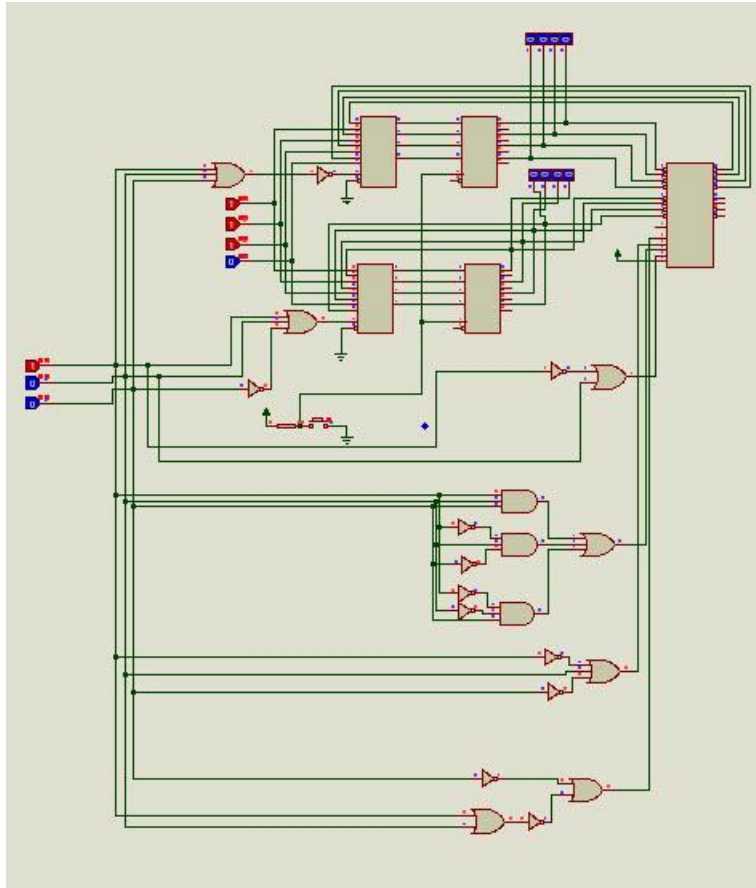
ابتدا در نرم افزار Proteus، در قسمت Devices دستگاه های مورد نیاز مانند گیت های AND، OR، LOGICSTATE، BUTTON، RESISTOR، NOT، LOGICTOGGLE، LOGICPROP و 74157 و 74181 و 74175 را از کتابخانه ها انتخاب میکنیم تا در پنجره ی سمت چپ قرار بگیرند. کلیت مدار را مطابق شکل در دستور کار میسازیم و سپس سیگنال های کنترلی را با توجه به عبارت های به دست آمده در قسمت قبل میسازیم.

نحوه ساخت کلاک مانند آزمایش دوم میباشد .

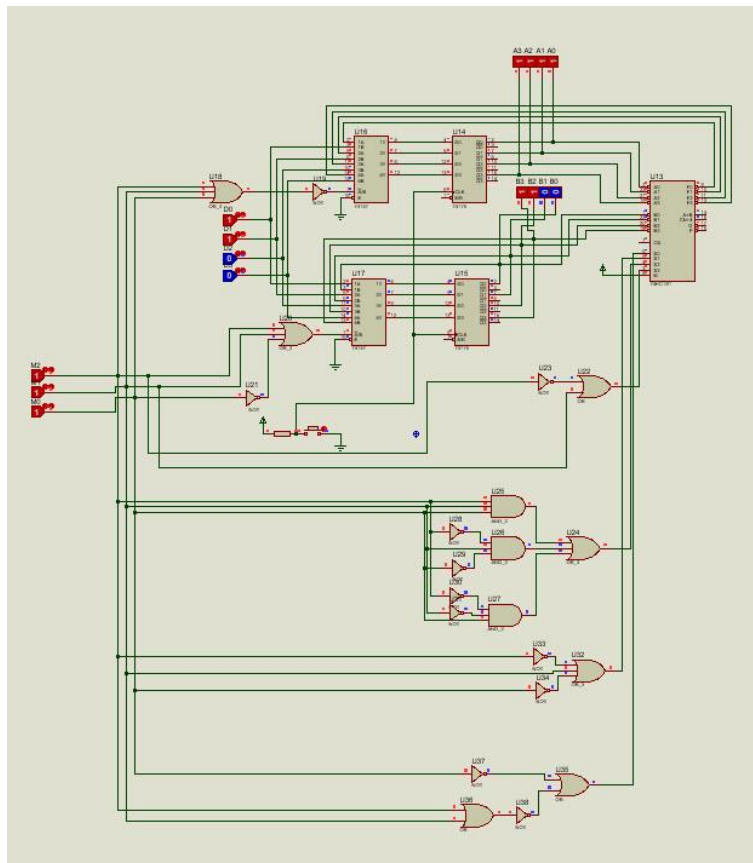
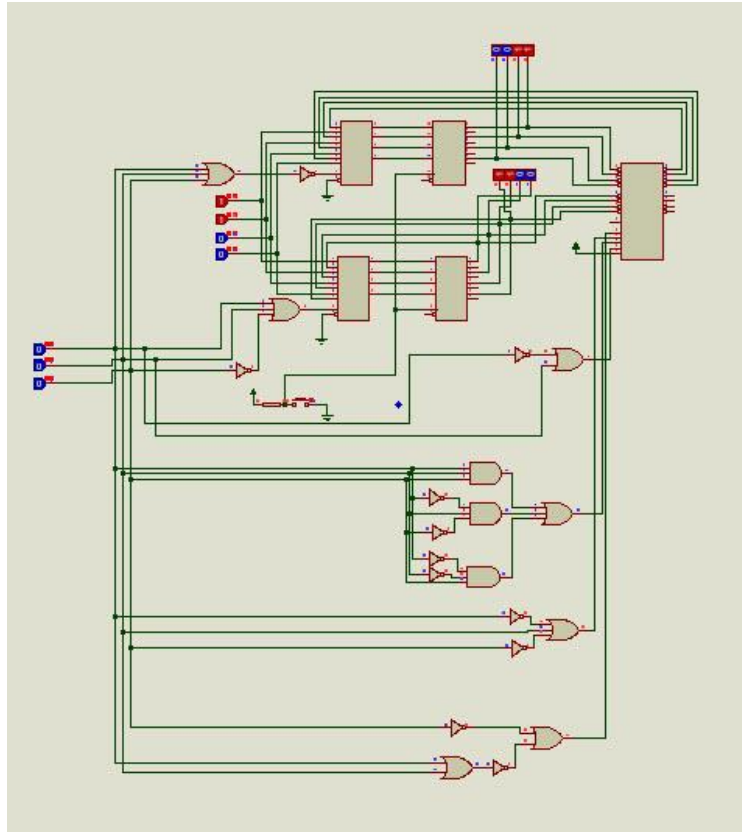


۱.۴ - امتحان اجزای مدار

در شکل بالا قرار داده ایم: $M_2M_1M_0 = 000$ و $D_3D_2D_1D_0 = 0111$ و سپس کلاک را اعمال کرده ایم. عدد ۷ در رجیستر A ذخیره می‌شود. سپس قرار می‌دهیم $M_2M_1M_0 = 100$ و کلاک را اعمال می‌کنیم رجیستر A ریست می‌شود.



در شکل زیر نیز عدد ۱۱۰۰ را در B و عدد ۰۰۱۱ را در A ذخیره کرده ورودی $M_2M_1M_0 = 111$ را می‌دهیم و می‌بینیم که عدد ۱۱۱۱ در A ذخیره می‌شود.



ساخت مدار داخلی ALU

در این بخش می‌خواهیم، یک واحد محاسبات و منطق چهاربیتی (4-bit ALU) طبق شکل زیر بسازیم.

Operation select					Operation	Function
S_3	S_2	S_1	S_0	C_{in}		
0	0	0	0	0	$F = A$	Transfer A
0	0	0	0	1	$F = A + 1$	Increment A
0	0	0	1	0	$F = A + B$	Addition
0	0	0	1	1	$F = A + B + 1$	Add with carry
0	0	1	0	0	$F = A + \overline{B}$	Subtract with borrow
0	0	1	0	1	$F = A + \overline{B} + 1$	Subtraction
0	0	1	1	0	$F = A - 1$	Decrement A
0	0	1	1	1	$F = A$	Transfer A
0	1	0	0	x	$F = A \wedge B$	AND
0	1	0	1	x	$F = A \vee B$	OR
0	1	1	0	x	$F = A \oplus B$	XOR
0	1	1	1	x	$F = \overline{A}$	Complement A
1	0	x	x	x	$F = shr A$	Shift right A into F
1	1	x	x	x	$F = shl A$	Shift left A into F

۲.۱ - تجزیه و تحلیل تیوری آزمایش

برای طراحی مدار به صورت زیر فکر میکنیم.

۸ خط اول را سعی میکنیم باهم انجام دهیم در همه ی آنها A با چیزی جمع شده است پس میتوان یک Adder گذاشت که یک ورودی آن A باشد و ورودی دیگر B یا not(B) و یا ۰ باشد و cin نیز ۱ یا ۰ باشد اینکه ورودی دیگر چه باشد را میتوان با استفاده از چند طبقه mux ۴ تایی درست کرد.

۴ خط بعد را نیز سعی میکنیم باهم انجام دهیم در همه ی آن ها خروجی یک عبارت است بر حسب A و B ، پس عبارات ممکن را ساخته و بعد توسط ۲ طبقه mux ۴ تایی مشخص میکنیم کدام یک مدنظر است.

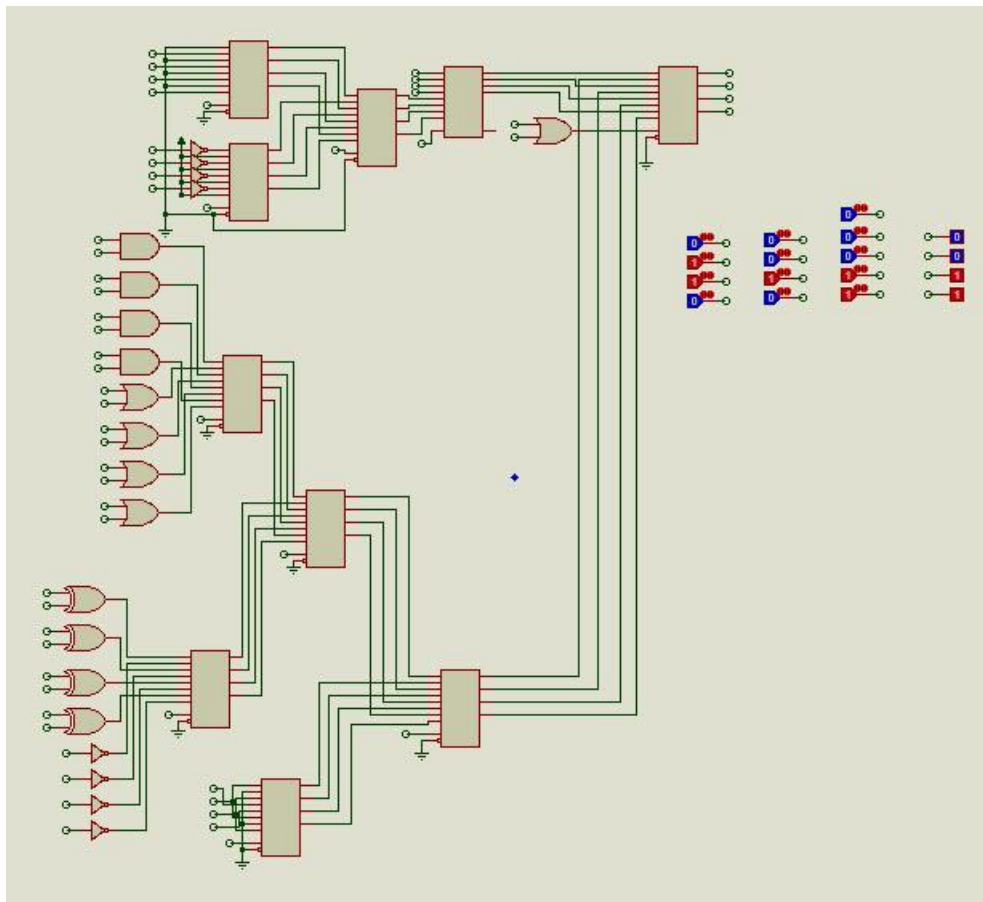
۲ خط آخر را نیز سعی میکنیم باهم انجام دهیم از یک mux ۴ تایی استفاده میکنیم که متناسب با شیفت به راست یا چپ ورودی های آن به بیت های صحیح و یا ۰ متصل شده اند.

اکنون با استفاده از دو mux ۴ تایی دیگر حاصل نهایی را پیدا میکنیم.

۲.۲ - نحوه ی ساخت مدار

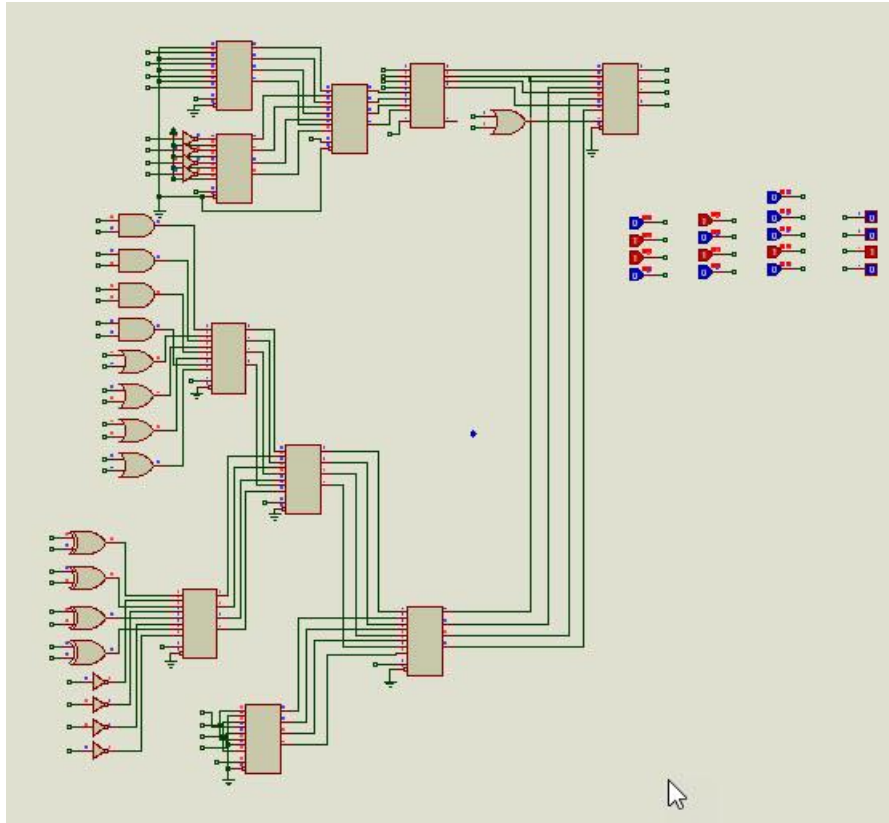
ابتدا در نرم افزار Proteus، در قسمت Devices دستگاه‌های مورد نیاز مانند گیت‌های AND، OR، LOGICPROP، LOGICTOGGLE، NOT، LOGICSTATE و 74157 و 4008 را از

کتابخانه‌ها انتخاب میکنیم تا در پنجره ی سمت چپ قرار بگیرند. حال مدار را مطابق توضیحات بخش تجزیه و تحلیل تیوری میسازیم.



۲.۳ - امتحان اجزای مدار

ورودی عدد A را $A_3A_2A_1A_0 = 0110$ و ورودی عدد B را $B_3B_2B_1B_0 = 0101$ می‌دهیم. ، بیت‌های سلکتور را نیز برابر $S_3S_2S_1S_0c_{in} = 01000$ قرار می‌دهیم. عدد ۰۱۰۰ در خروجی نمایش داده می‌شود که صحیح بوده و برابر and دو ورودی میباشد.



منابع و مراجع

[1] Mano, M. Morris. Computer system architecture. Prentice-Hall of India, 2003.